



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/816,615	04/02/2004	Ashok Banerjee	BEAS-01485US0	9803
23910	7590	01/09/2008	EXAMINER	
FLIESLER MEYER LLP			INGBERG, TODD D	
650 CALIFORNIA STREET			ART UNIT	
14TH FLOOR			PAPER NUMBER	
SAN FRANCISCO, CA 94108			2193	
			MAIL DATE	DELIVERY MODE
			01/09/2008	PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

## Office Action Summary

Application No.

10/816,615

Applicant(s)

BANERJEE ET AL.

Examiner

Todd Ingberg

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 05 April 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-25 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-25 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 4/5/04 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_.
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_.

### **DETAILED ACTION**

Claims 1 – 25 have been examined.

#### ***Drawings***

1. The drawings filed April 2, 2004 have been accepted.

#### ***Specification***

2. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.
3. Claims 7, 9, 11, 19, 21 and 23 are objected to because of the following informalities: the word “wherein” appears duplicate. Appropriate correction is required.

#### ***Claim Rejections - 35 USC § 102***

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1 – 2, 4 – 14, 16 – 25 are rejected under 35 U.S.C. 102(b) as being anticipated by Rational Software Corporations product Rational PureCoverage from 2001.

#### **Claim 1**

A system for software source code analysis, comprising: a software interface to a source code management system, which can be used to identify changes that have occurred to source code over a specified interval of time or relative changes, independent of any particular source code management tool implementation; a software interface to a code coverage database, which can be used to identify what source code has been exercised during a test run, independent of any particular code coverage tool implementation; a software interface for identifying which source files are exercised in a software product by a particular software test, using code coverage

Art Unit: 2193

mapping, a software interface for identifying what tests have failed during a test execution cycle for a particular software product, independent of any particular testing technology or test execution tool; and a bug inspection analyzer that determines the failure-to-change intersection point by integrating the information culled from the software interfaces described above to determine what tests failed, what source code files of the target software product are exercised by the failing tests, and which of the files identified thereby have been changed by a product software developer since the failed tests were last known to be in a passing state or within some other specified timeframe.

**Examiner's Rejection**

Pure anticipates a system for software source code analysis (Pure Coverage – product, page 1), comprising: a software interface to a source code management system (Pure, page 2, bottom – merge), which can be used to identify changes that have occurred to source code over a specified interval of time or relative changes (Pure, page 3, top line – Analysis-time mode options), independent of any particular source code (Pure, page 2 – separate commands) management tool implementation (Pure, page 2 – Run-time options); a software interface to a code coverage database, which can be used to identify what source code has been exercised during a test run, independent of any particular code coverage tool implementation (Pure, page 2, bottom – merge); a software interface for identifying which source files are exercised in a software product by a particular software test (Pure, page 4, API functions), using code coverage mapping (Pure, page 4, pc\_select), a software interface for identifying what tests have failed during a test execution cycle for a particular software product, independent of any particular testing technology or test execution tool (Pure, page 1 – PureCoverage – bottom screen see “Mark adjustments tells of types of failures); and a bug inspection analyzer that determines the failure-to-change intersection point by integrating the information culled from the software interfaces described above to determine what tests failed (Pure, pages 2 – 3, Analysis-time options), what source code files of the target software product are exercised by the failing tests (Pure, page 5, Running a make-run-debug-edit-cycle), and which of the files identified thereby have been changed by a product software developer since the failed tests were last known to be in a passing state **OR** within some other specified timeframe (Pure, page 3, top line – Analysis-time mode options).

**Claim 2**

The system of claim 1 wherein changes to the software code between said first version and said second version are identified by change label.

**Examiner's Rejection**

Pure, page 4, diff commands in Report scripts.

**Claim 4**

The system of claim 1 wherein the code coverage interface includes an input for allowing an operator to specify either of date and/or code change ranges to be analyzed by said bug inspection analyzer.

**Examiner's Rejection**

Pure, page 2 – Analysis-time options and page 5 – Running a make-run-debug-edit cycle.

**Claim 5**

Art Unit: 2193

The system of claim 1 wherein the interfaces are realized to interact with a tool-specific implementation that interfaces to a vendor-specific subsystem

**Examiner's Rejection**

Interpreted to be Rational Software's very well known tools for software development.

Page 4 is the API support. The source code in the examples would be the product of the tools.

**Claim 6**

The system of claim 1 wherein the source code interface includes an interface to a vendor-specific SCM system.

**Examiner's Rejection**

Pure, page 1, bottom screen – Pure Coverage Annotated Source Window.

**Claim 7**

The system of claim 6 wherein wherein the system includes said SCM system.

**Examiner's Rejection**

See the rejection for claim 6.

**Claim 8**

The system of claim 1 wherein the test interface includes an interface to a TER system.

**Examiner's Rejection**

Pure, page 1, top screen – see controls for functionality related to screen shot.

**Claim 9**

The system of claim 8 wherein wherein the system includes said TER system.

**Examiner's Rejection**

See the rejection for claim 8.

**Claim 10**

The system of claim 1 wherein the test interface includes an interface to a code testing system.

**Examiner's Rejection**

Pure, page 1, screen shots.

**Claim 11**

The system of claim 8 wherein wherein the system includes said code testing system.

**Examiner's Rejection**

See the rejection for claim 10.

**Claim 12**

A system for software source code analysis, comprising: means for retrieving a software code and running test suites against it at a first time and at a second time; means for importing code coverage data into the framework for failure analysis; means for retrieving detailed set of line-level product changes from a source code management system; and, means for comparing line-level code coverage data for a test case from a code coverage toolset to line-level change

Art Unit: 2193

information from a source code management system, and determining an intersection of these two data sets to represents the set of critical changes over the specified time period.

**Examiner's Rejection**

Pure anticipates a software source code analysis (Pure Coverage – product, page 1), comprising: means for retrieving a software code (Pure, page 2, bottom – merge) and running test suites against it at a first time and at a second time; means for importing code coverage data into the framework for failure analysis (Pure, page 3, top); means for retrieving detailed set of line-level product changes from a source code management system (Pure, page 4, pc\_diff – difference); and, means for comparing line-level code coverage data for a test case from a code coverage toolset to line-level change information from a source code management system (diff as per above), and determining an intersection of these two data sets to represents the set of critical changes over the specified time period (Pure, page 3, top line – Analysis-time mode options).

**Claim 13**

A method for software source code analysis, comprising the steps of: accessing a source code management system, which can be used to identify changes that have occurred to source code over a specified interval of time or relative changes, independent of any particular source code management tool implementation; accessing a code coverage database, which can be used to identify what source code has been exercised during a test run, independent of any particular code coverage tool implementation; identifying which source files are exercised in a software product by a particular software test, using code coverage mapping; identifying what tests have failed during a test execution cycle for a particular software product, independent of any particular testing technology or test execution tool; and determining the failure-to-change intersection point by integrating the information culled from the software interfaces described above to determine what tests failed, what source code files of the target software product are exercised by the failing tests, and which of the files identified thereby have been changed by a product software developer since the failed tests were last known to be in a passing state or within some other specified timeframe. See the rejection for claim 1.

**Claim 14**

The system of claim 13 wherein changes to the software code between said first version and said second version are identified by change label. See the rejection for claim 2.

**Claim 16**

The system of claim 13 wherein the code coverage interface includes an input for allowing an operator to specify either of date and/or code change ranges to be analyzed by said bug inspection analyzer. See the rejection for claim 4.

**Claim 17**

The system of claim 13 wherein the interfaces are realized to interact with a tool-specific implementation that interfaces to a vendor-specific subsystem. See the rejection for claim 5.

**Claim 18**

Art Unit: 2193

The system of claim 13 wherein the source code interface includes an interface to a vendor-specific SCM system. See the rejection for claim 6.

**Claim 19**

The system of claim 18 wherein wherein the system includes said SCM system.  
See the rejection for claim 7.

**Claim 20**

The system of claim 13 wherein the test interface includes an interface to a TER system.  
See the rejection for claim 8.

**Claim 21**

The system of claim 20 wherein wherein the system includes said TER system.  
See the rejection for claim 9.

**Claim 22**

The system of claim 13 wherein the test interface includes an interface to a code testing system.  
See the rejection for claim 10.

**Claim 23**

The system of claim 20 wherein wherein the system includes said code testing system.  
See the rejection for claim 11.

**Claim 24**

A method of software source code analysis, comprising the steps of: retrieving a software code and running test suites against it at a first time and at a second time; importing code coverage data into the framework for failure analysis; retrieving detailed set of line-level product changes from a source code management system; and comparing line-level code coverage data for a test case from a code coverage toolset to line-level change information from a source code management system, and determining an intersection of these two data sets to represents the set of critical changes over the specified time period. See the rejection for claim 12.

**Claim 25**

A computer readable medium including instruction stored thereon which when executed cause the computer to perform the steps of: accessing a source code management system, which can be used to identify changes that have occurred to source code over a specified interval of time or relative changes, independent of any particular source code management tool implementation; accessing a code coverage database, which can be used to identify what source code has been exercised during a test run, independent of any particular code coverage tool implementation; identifying which source files are exercised in a software product by a particular software test, using code coverage mapping; identifying what tests have failed during a test execution cycle for a particular software product, independent of any particular testing technology or test execution tool; and determining the failure-to-change intersection point by integrating the information culled from the software interfaces described above to determine what tests failed, what source

Art Unit: 2193

code files of the target software product are exercised by the failing tests, and which of the files identified thereby have been changed by a product software developer since the failed tests were last known to be in a passing state or within some other specified timeframe.  
See the rejection for claim 1.

***Claim Rejections - 35 USC § 103***

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 3 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pure as applied to claim 1 above, and further in view of Web Management with Microsoft Visual SourceSafe 5.0, 1997 (VSS).

**Claim 3**

The system of claim 1 wherein changes to the software code between said first version and said second version are identified by modification date.

**Examiner's Rejection**

Pure teaches analysis of software and building software (create a new version). But Pure does not explicitly mention the well known technique of time stamping with check in and check out (Pure, page 5 – check in and out). It is VSS, who teaches time stamping (VSS, page 101 - \$DATE: \$). Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to combine Pure and VSS because keyword labels on version help identify the history of the software builds.

**Claim 15**

The system of claim 13 wherein changes to the software code between said first version and said second version are identified by modification date. See the rejection for claim 3.

***Correspondence Information***

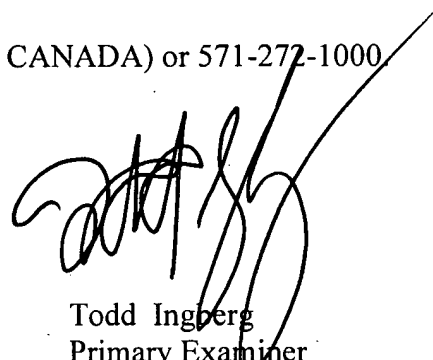
Any inquiry concerning this communication or earlier communications from the examiner should be directed to Todd Ingberg whose telephone number is (571) 272-3723. The examiner can normally be reached on during the work week..



Art Unit: 2193

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.



Todd Ingberg  
Primary Examiner  
Art Unit 2193

TI